WELLS FARGO

Conceptual Soundness: Model Explainability

Sep 25, 2023

Linwei Hu

Corporate Model Risk
Wells Fargo

Outline

- Introduction: model soundness
- Model Explainability
  - Post hoc techniques for model explanation
  - Inherently-interpretable ML algorithms (next class)
- Illustrative Example
- Summary

Model soundness

- Make sense
  - Model and results are interpretable and can be explained to stakeholders
  - Results are consistent with subject-matter expertise
- Good predictive performance
  - In comparison to other algorithms (benchmark models for high-risk rank)
  - Global as well as local
  - Generalizable to potential new environments (*when it should*)
    - Stable to certain changes when it should be
    - Good sensitivity to certain changes in key predictors (stress test)
- Robust
  - Not overly flexible and unstable
  - Does not overfit training data (globally or locally)
- Fair
  - Does not discriminate based on protected attributes
  - Results are fair to customers and other stakeholders
- Above points are not mutually exclusive

Opportunities and Challenges with ML

- Good predictive performance, works well with large datasets
  - ➢ Flexible modeling
  - ➢ Automated approach to feature engineering
    - ○ Saves time
    - ○ Useful in new applications with insufficient prior knowledge on feature engineering
- BUT … Predictor $\hat{f}(x)$ is implicitly defined, high-dimensional, and complex
  - – Hard to interpret results
  - – Not an issue if goal is only prediction: recommender systems, fraud detection, …
  - – Big issue for regulated industries and safety-critical applications
  - – Banks have dual goals: good predictive performance and ensure results make sense
- Must be able to explain the model, results, and develop insights
    - ○ Why? Provide explanations to multiple stakeholders
      - ✓ Make sure model make sense → consistent with subject-matter knowledge
      - ✓ For certain applications, model must be “fair”

Outline

- Introduction: conceptual soundness
- Model Explainability
  - Post hoc techniques for model explanation
  - Inherently-interpretable ML algorithms (next class)
- Illustrative Example
- Summary

Making sense: Understanding model results

Main approaches:

I. Post hoc: Techniques for interpreting results after fitting the ML algorithm
   a) Global – Important predictors; input-output relationships
   b) Local – how does model behave locally; contribution of predictors to a particular prediction

II. Inherently interpretable algorithms (next class)
   a) Low-order functional ANOVA models
   b) Additive index models

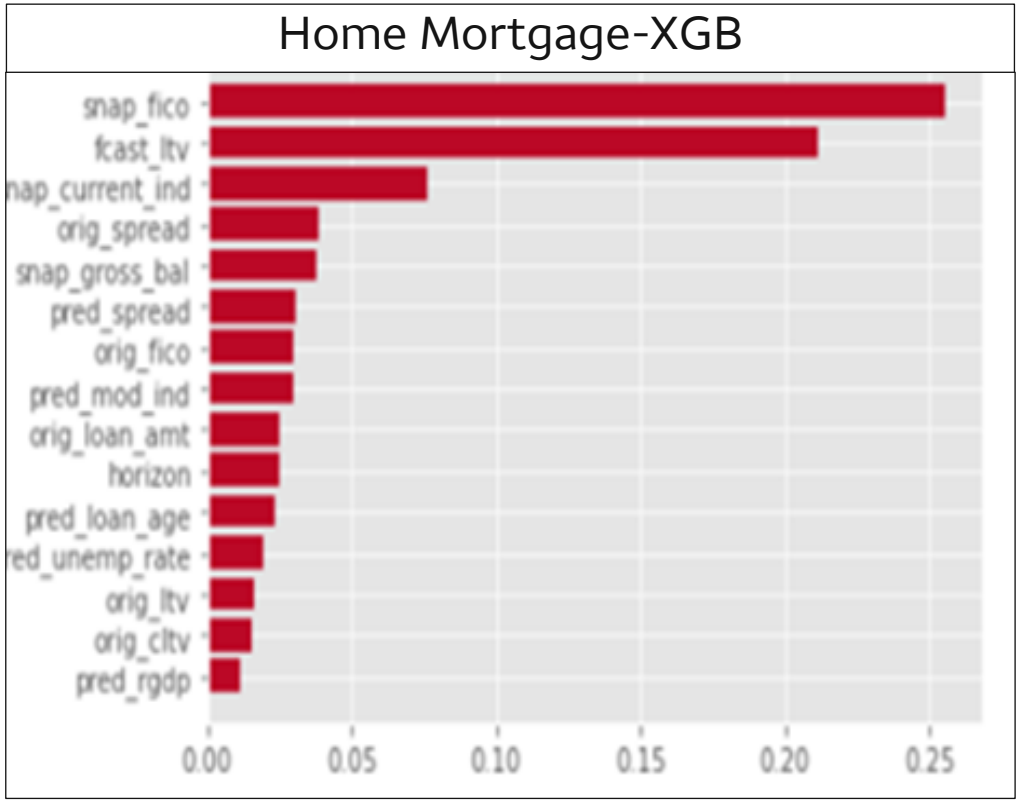
III. Fitting and using surrogate models to explain complex results (skip)
   a) Born-again trees (piecewise constant) → Breiman
   b) Surrogate locally interpretable model → Hu, Chen, Nair (2022)

Post hoc global: Identifying important predictors/features

- **Permutation based: Model agnostic**
  - Randomly permute the rows for variable (column) of interest while keeping everything else unchanged
  - Get the model predictions for the permuted data
  - Compute the change in prediction performance as the measure of importance.
  -

Y	X1	X2	Xj -> Xj_perm	X4	X5
2	1.5	0	4.5 -> 5.3	10.2	3.0
4	2.7	1	5.3 -> 3.3	8.7	4.2
8	3.3	1	7.2 -> 4.5	19.3	17.6
3	1.9	0	3.3 -> 7.2	7.8	21.2

- **Selected Others** (Many in literature)
  - **Tree-based** importance metrics
    - Importance of a variable x_j based on impurity
      - Total reduction of impurity at all nodes where x_j used for splitting
    - For ensemble algorithms, average over all trees
  - **Global Shapley**
    - Based on Shapley decomposition (1953);
    - Owen (2014) and others applied it to ML feature importance
    - Model agnostic but **computationally intractable**



Understanding input-output relationships: 1-dimensional partial dependence plots

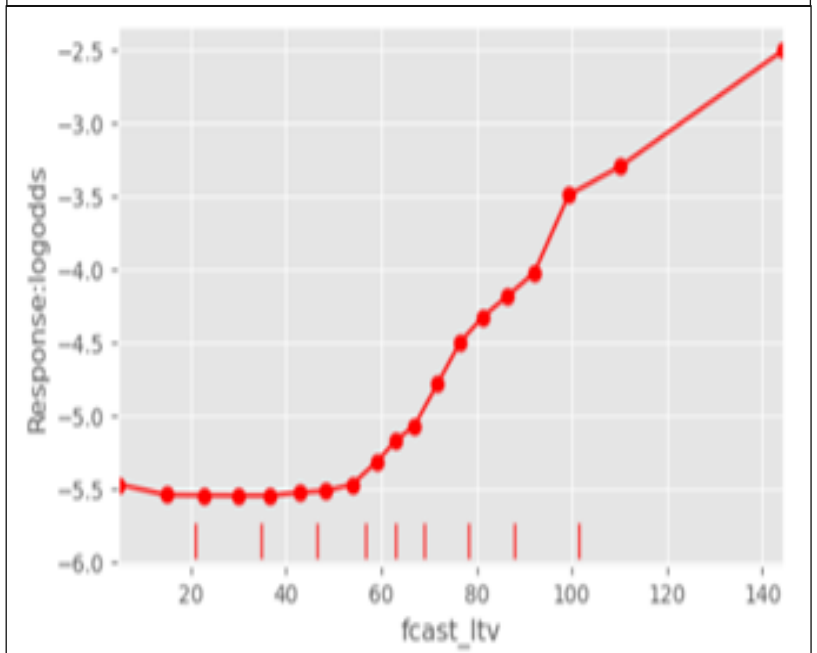
- Understand how fitted response varies as a function of the variable of interest
- **One-dimensional Partial Dependence Plot (PDP)**
  - Friedman, J. H (2001). Greedy Function Approximation: A Gradient Boosting Machine.
  - Variable of interest: x_j
  - Write the fitted model as $\hat{f}(x) = \hat{f}(x_j, \boldsymbol{x}_{-j})$
  - Fix x_j at c; compute the average of $\hat{f}$ over the entire data

$$f_{pdp,j}(x_j = c) = \frac{1}{N}\sum_{i=1}^{N} \hat{f}(x_j = c, x_{-j,i}), i = 1, \ldots, N$$

  - Plot $f_{pdp,j}(x_j)$ against x_j over a grid of values $c_1, \ldots\ c_m$
  - One-dimensional summary
  - Interpretation: Effect of x_j averaged over other variables

Y	X1	X2	Xj	X4	X5
2	1.5	0	c	10.2	3.0
4	2.7	1	c	8.7	4.2
8	3.3	1	c	19.3	17.6
3	1.9	0	c	7.8	21.2

Home Mortgage
1-D PDP for Forecasted LTV



Assessing interactions

I. ICE (individual conditional expectation) plots
   Goldstein, A., Kapelner, A., Bleich, J., & Pitkin, E. (2013). Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation. *eprint arXiv:1309.6392*

II. Two-dimensional partial dependence plots

III. H-statistics for quantifying two-dimensional interactions
   Friedman, J. H (2001). Greedy Function Approximation: A Gradient Boosting Machine. The Annals of Statistics, 29 (5): 1189-1232.

- One can use ICE plots to detect the presence of interactions, but they do not give further insights
- Items II and II examine interactions with specific pairs of variables
- They can be extended to higher-dimensional interactions

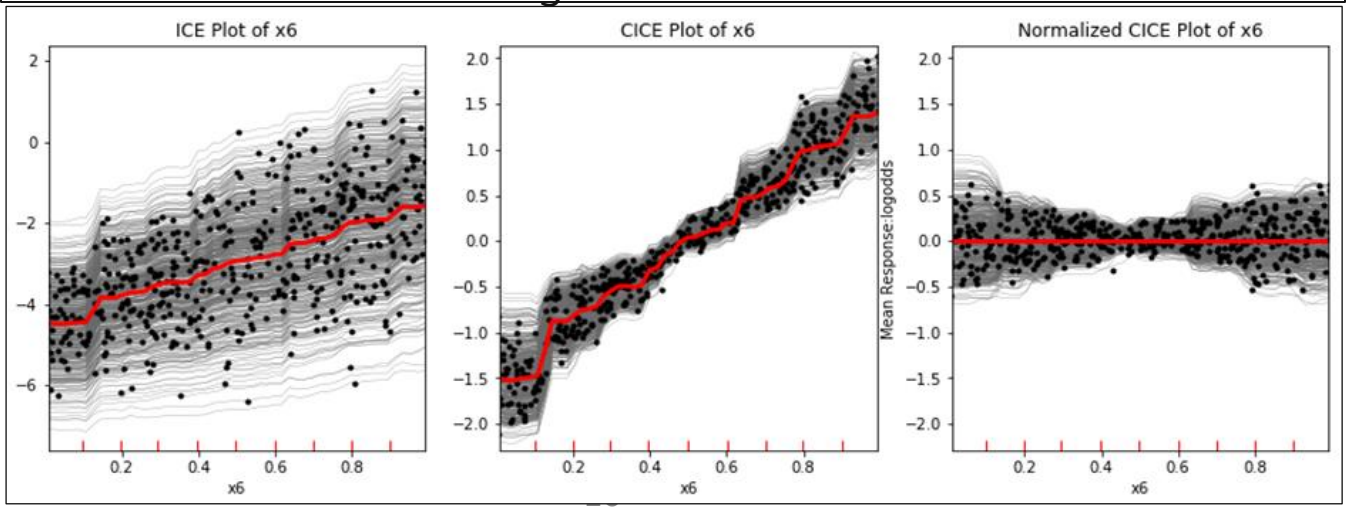
Individual Conditional Expectation Plots

- 1-d partial dependence plot $f_{pdp,j}(x_j)$ shows the average over the entire data

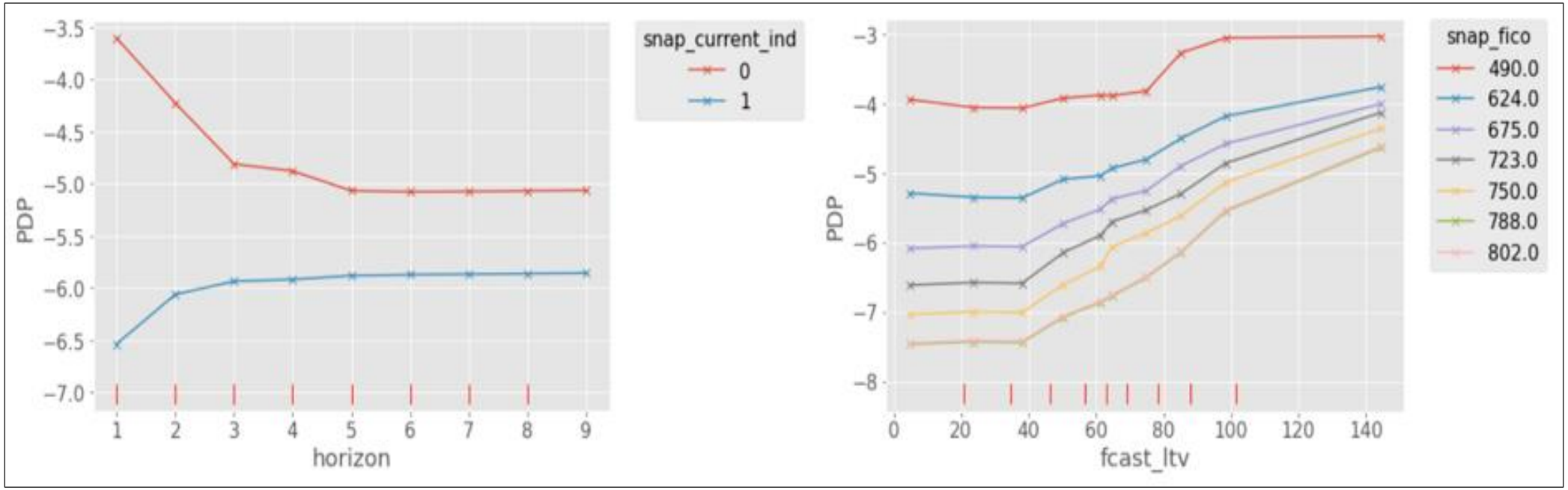
$$f_{pdp,j}(x_j) = \frac{1}{N}\sum_{i=1}^{N} \hat{f}(x_j, x_{-j,i})$$

- When there are interaction effects, $\hat{f}(x_j, x_{-j,i})$ will have different patterns for different $x_{-j,i}$.
- So averaging will lose the interaction information.
- The ICE plot is a plot of all the N curves $\hat{f}(x_j, x_{-j,i})$ against x_j, $i = 1, 2, \ldots, N$
- Each curve is localized for a single ith observation.
- It allows us to see if there is any change of the input-output relationships for x_j, thus to see any interaction effect.

- ICE plot for a simulation example.
  - True model: $log\left(\frac{p_i}{1-p_i}\right) = -8 + 1.6\,x_1 + 4\sqrt{x_2} - x_3^2 + x_5^2 + 2.4\,x_6 + 2x_1x_6$
  - The black curves are the ICE curves over a grid of x_6.
  - The dots show the observed value of x_6 for ith observation.
  - The red curve is the PDP, which is the average of all ICE curves.
  - CICE plots are centered versions
  - Parallel curves → no interaction. Otherwise, there is interaction
  - It shows x_6 has interaction effects since the ICE curves are not parallel. However, we don't know (from ICE plot) which variable it is interacting with.



Input-output relationships: Two-dimensional partial dependence plots



- 2D-PDPs to show pairwise interaction of variables x_j, x_k
- Fix $x_j = c, x_k = d$ and compute average of model prediction
- Multiple curves: Each curve fixes value of the second variable and change values for the first variable
- Non-parallel curves show interactions: horizon and current indicator, fico and ltv both show interaction

Y	X1	X2	Xj	Xk	X5
2	1.5	0	c	d	3.0
4	2.7	1	c	d	4.2
8	3.3	1	c	d	17.6
3	1.9	0	c	d	21.2

H-statistics to measure two-dimensional interactions

- H_{jk} to measure the interaction between x_j and x_k, by quantifying the degree of non-parallelism in 2d PDP.

$$H_{jk}^2 = \frac{\sum_{i=1}^{N}\left[f_{pdp}(x_{ij},x_{ik}) - f_{pdp}(x_{ij}) - f_{pdp}(x_{ik})\right]^2}{\sum_{i=1}^{N} f_{par}^2(x_{ij},x_{ik})}, \; H_{jk} = \sqrt{H_{jk}^2}$$

  - $f_{pdp}(x_{ij}, x_{ik}), f_{pdp}(x_{ij}), f_{pdp}(x_{ik})$ are the centered two and one-dimensional partial dependence functions
  - H_{jk}^2 is the proportion of variation in $f_{par}(x_{ij}, x_{ik})$ unexplained by an additive model – relative measure
  - There is no easy way assess if it is large or small

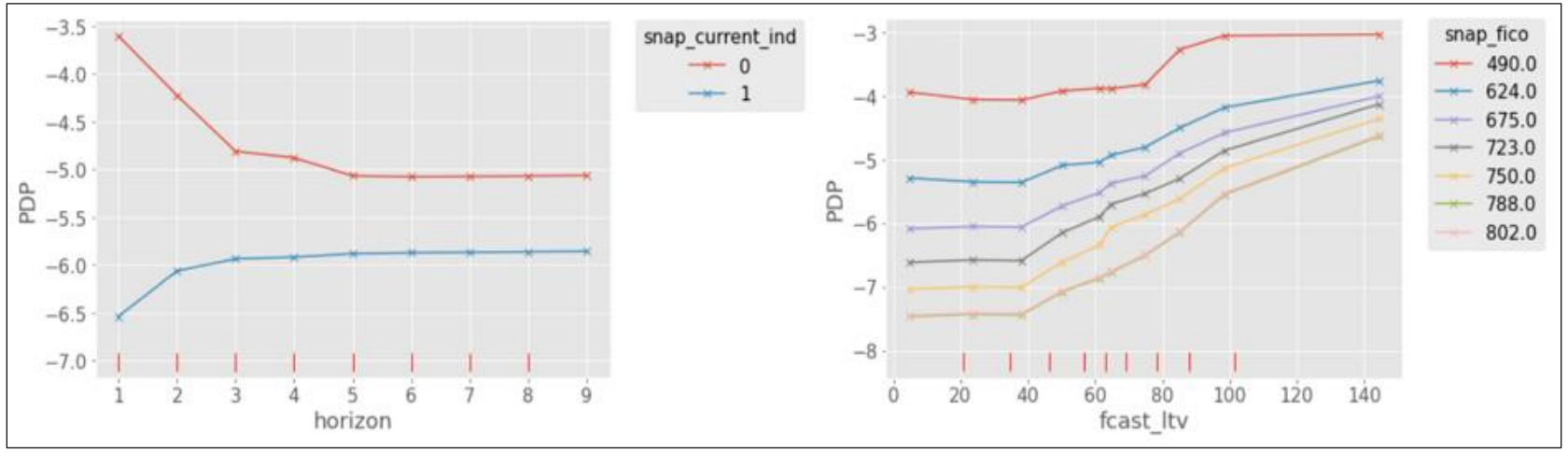
- One can also use an absolute version of H-statistic (without the denominator)

$$\widetilde{H}_{jk}^2 = \frac{1}{N}\sum_{i=1}^{N}\left[f_{pdp}(x_{ij}, x_{ik}) - f_{pdp}(x_{ij}) - f_{pdp}(x_{ik})\right]^2, \; \widetilde{H}_{jk} = \sqrt{\widetilde{H}_{jk}^2}$$

Illustration: 2-D PDPs and H-statistics: Home Lending Example

- Interactions between FICO and LTV_forcast (left), h and dlq_new_clean(right).

	snap_fico	fcast_ltv	snap_current_ind	unemprt	totpersincyy	horizon	premod_ind
snap_fico	NaN	0.163	0.1224	0.082	0.036	0.0339	0.1107
fcast_ltv	0.163	NaN	0.0518	0.0291	0.0286	0.0186	0.0843
snap_current_ind	0.1224	0.0518	NaN	0.0101	0.0071	0.2296	0.0003
unemprt	0.082	0.0291	0.0101	NaN	0.0232	0.0122	0.0094
totpersincyy	0.036	0.0286	0.0071	0.0232	NaN	0.0068	0.0661
horizon	0.0339	0.0186	0.2296	0.0122	0.0068	NaN	0.0192
premod_ind	0.1107	0.0843	0.0003	0.0094	0.0661	0.0192	NaN



Delinquency vs horizon

LTV_fcast vs fico

Correlation can create havoc!

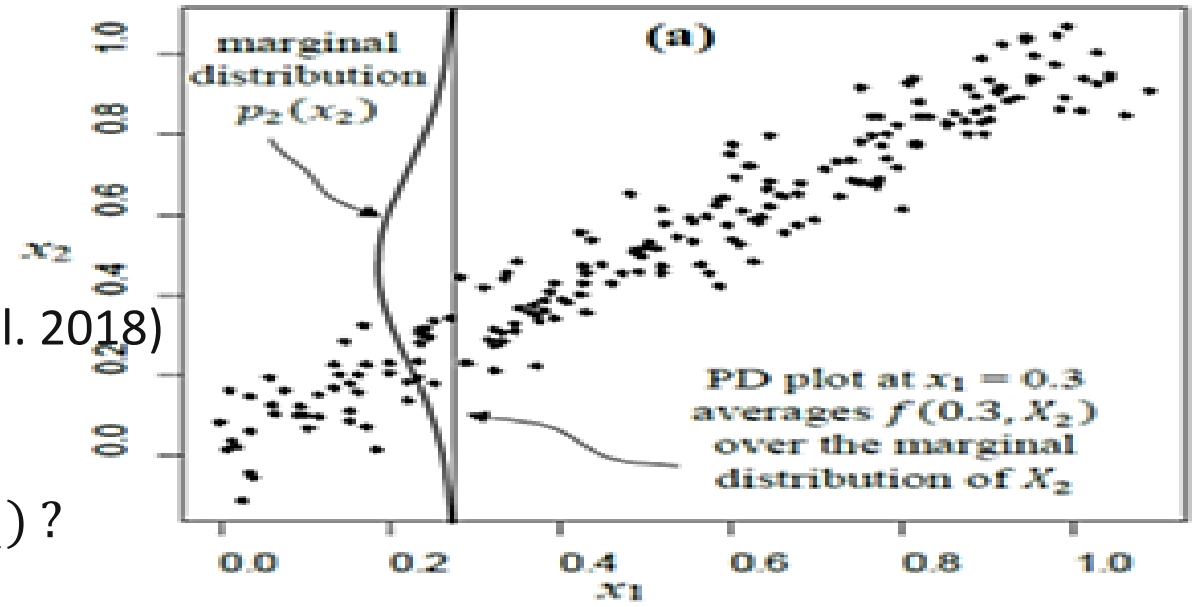
$\hat{f}(\boldsymbol{x}) = \hat{f}(\boldsymbol{x_j}, \boldsymbol{x_{-j}})$ is the fitted model

$$\hat{f}_{PD,j}(z) = \frac{1}{N}\sum_{i=1}^{N} \hat{f}(x_j = z, \boldsymbol{x_{-j,i}})$$

When predictors are highly correlated:

Performance of VI analyses and PDPs?

- Extrapolation
- Poor model fit outside data envelope
- Alternatives: ALE (Apley and Zhu, 2020), ATDEV (Liu et al. 2018)



Bigger issue: Model identifiability

$$f(x_1, x_2) = \beta_1 x_1 + \beta_2 x_2 + \beta_{12}\, x_1 x_2 \rightarrow g(x_1)\ ?$$

- Hard to tell the difference from $x_1 x_2$ or x_1^2
- Different ML algorithms can capture $x_1 x_2$ differently
- VI analysis → permute correlated variables jointly

These are known problems to statisticians → that's why there has been a lot of model diagnostics!
But the view in ML is to throw as many predictors as possible into the mix and automate model building

No easy answers!

Techniques for local explanation

- **Two questions of Interest:**

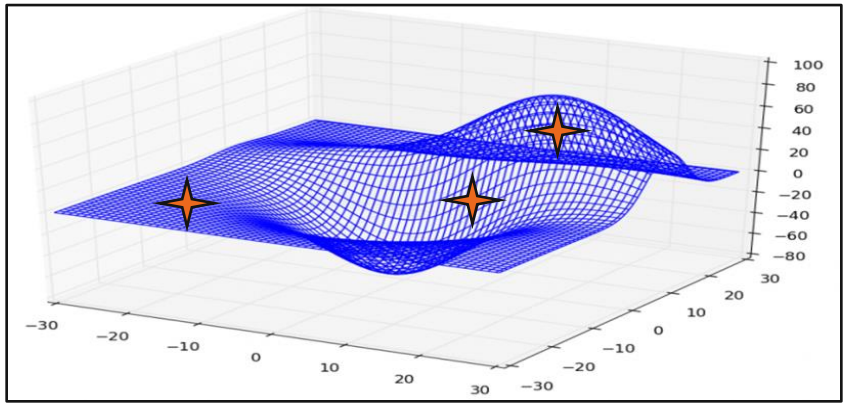
  1. How does the model behave locally at a point of interest?
  2. Consider the predicted value at a point of interest $\hat{f}(\boldsymbol{x}^*) = \hat{f}(x_1^*, \ldots, x_K^*)$:
     What are the contributions of the different variables/features $\{x_1, \ldots x_K\}$ to this prediction?
     We will see an example of this in credit applications.

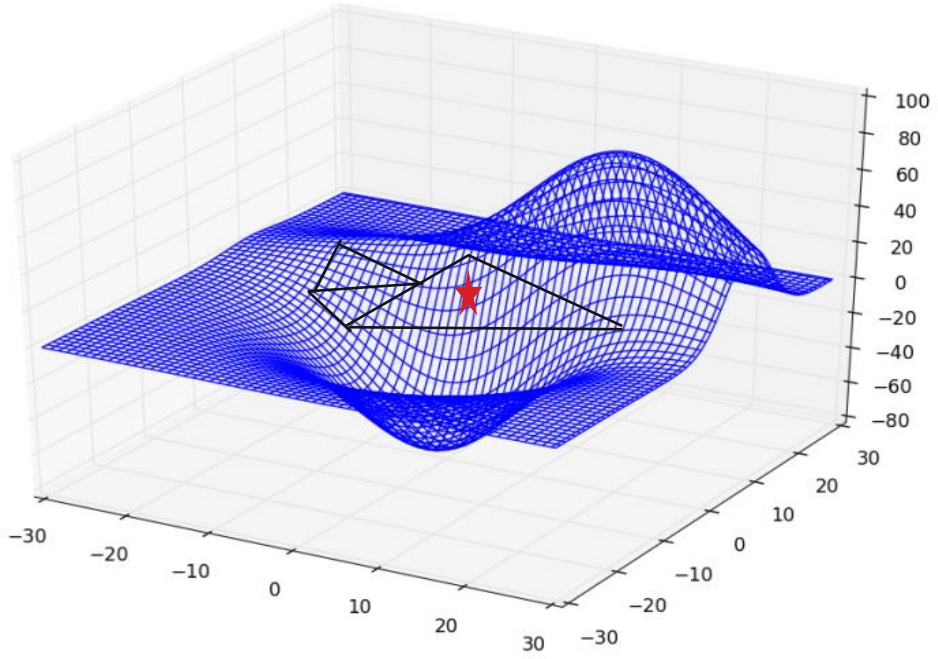
- If fitted model is linear: $\hat{f}(\boldsymbol{x}) = b_0 + b_1 x_1 + \ \ldots b_K x_k$, we can answer both questions using the regressions coefficients.
  - Answer to 1: Model is linear → magnitudes and signs of regression coefficients provide explanation
  - Answer to 2: Contribution of x_j^* is $b_j x_j^*$

- BUT … how to extend these interpretations to ML algorithms?

Techniques for local explanation: Question 1

- How can we interpret the response surface locally at selected points of interest?
- LIME(Local Interpretable Model agnostic Explanations.)
  - Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you?: Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (pp. 1135-1144)
  - Fit a linear model locally around the point and use for interpretation
  - Above reference suggests a particular way to fit a local model – but there are simple ways to do it.
- FFNNs based on RELU activation function
  - Essentially partitions predictor space into regions and fits a linear regression model within
  - Difference with LIME: local linear model developed for each point
  - FFNN-RELU yields same linear model for all points in region
- Piecewise constant tree
  - Single regression tree or RF or XGB
  - splits into rectangular regions





Techniques for local explanation: Question 2

- Consider the predicted value at a point of interest $\hat{f}(\boldsymbol{x}^*) = \hat{f}(x_1^*, \ldots, x_K^*)$
- What are the individual contributions of $\{x_1, \ldots x_K\}$ to this prediction?
- This is a very important question in explaining adverse actions (AA) on a customer → AA reason code
- AA occurs in different contexts: lending, insurance, job application, etc.
- Credit Lending:
  - **Regulation B of Equal Credit Opportunity Act** – Both consumers and businesses
    - **A refusal to grant credit in substantially the amount or terms requested in an application** … ;
    - **A termination of an account** or an **unfavorable change in the terms of an account** … ;
    - A **refusal to increase the amount of credit** available to an applicant …
- Applicants are **legally entitled to get an explanation** for a negative decision. Examples of AA Reason Codes

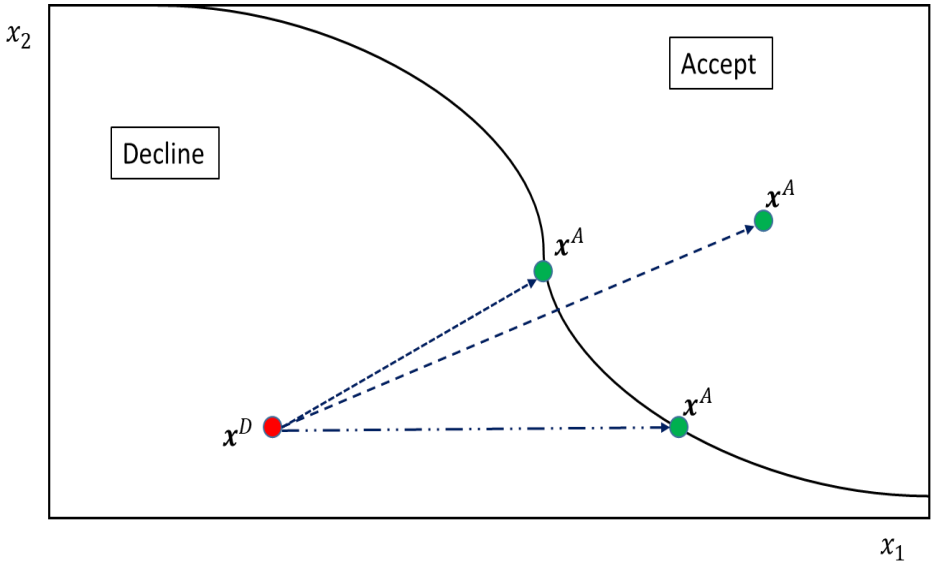
Issues with application	After assessment
• Credit application incomplete • Unable to verify credit references • Length of employment	• Insufficient income • Limited credit experience • Number of recent inquiries on credit bureau report • Delinquent credit obligations • Value of collateral not sufficient

Techniques for local explanation: Question 2

- If a bank's internal credit scoring model, $\hat{f}(\boldsymbol{x}^*) = \hat{f}(x_1^*, \ldots, x_K^*)$, predicts high default probability for an applicant, which results in a denial in credit application (Adverse Action), what are the reasons?
- In other words, what are the individual contributions of $\{x_1, \ldots x_K\}$ to this prediction?
- Involves comparison of prediction to a reference point ("average")
- Approaches in previous slides **cannot** be used
- Many techniques in the literature
- Common approaches based on local Shapley decomposition (called SHAP)
  - Lundberg and Lee (2017)
  - Many variations: Kernel SHAP, Tree SHAP
  - Recommend Baseline SHAP
  - Sundarajan and Najmi (2019)

General expression for B-SHAP

- Let $f(\boldsymbol{x})$ be the fitted model
- Consider two points in the predictor space: point of interest $\boldsymbol{x^D}$ (declined application) and a reference point $\boldsymbol{x^A}$
- Choice of reference points in "accept" space
  - Internal point: representing the average profile of accepted applications
  - On the boundary: the shortest distance for a declined application to be approved. Varies with declined application



- The goal is to decompose the difference $[f(\boldsymbol{x^D}) - f(\boldsymbol{x^A})]$ and attribute it to the difference variables $(x_1, \ldots x_K\}$
- Baseline-SHAP decomposition

$$E_k = E_k(\boldsymbol{x^D}; \boldsymbol{x^A}) = \sum_{\boldsymbol{S_k} \subseteq \boldsymbol{K} \setminus \{k\}} \frac{|\boldsymbol{S_k}|!\,(|\boldsymbol{K}| - |\boldsymbol{S_k}|)!}{|\boldsymbol{K}|!} \left(f\left(x_k^D;\ \boldsymbol{x^D_{S_k}}; \boldsymbol{x^A_{K \setminus S_k}}\right) - f\left(x_k^A; \boldsymbol{x^D_{S_k}}; \boldsymbol{x^A_{K \setminus S_k}}\right) \right).$$

- Looks formidable!

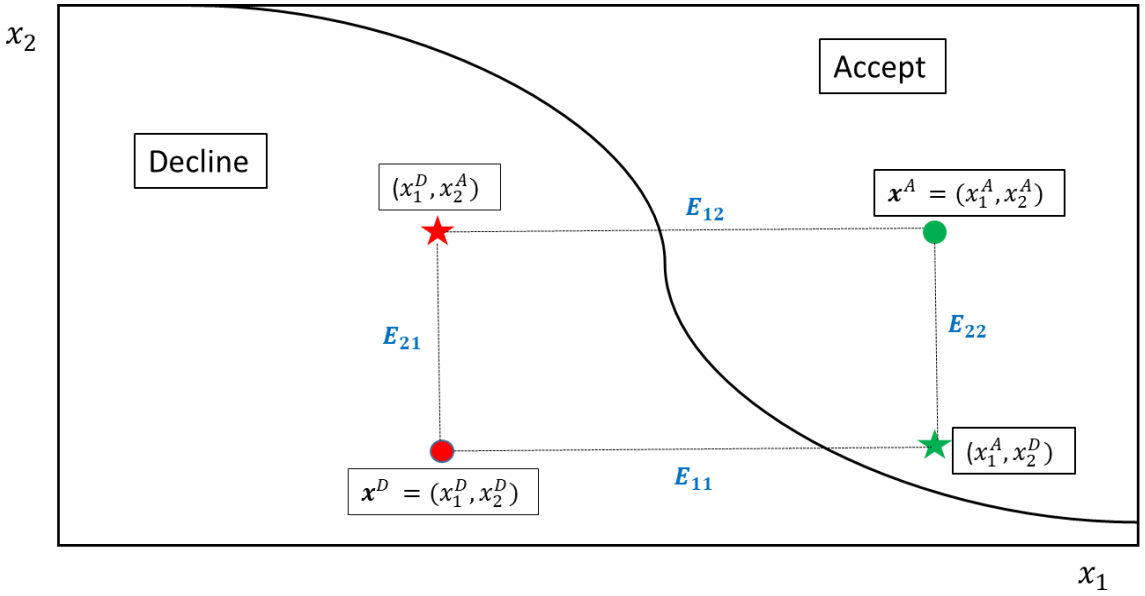
Case with two predictors: Motivation from first principles

- Consider simple case with two variables: $K = 2$
- The difference can be decomposed in 2 ways:

$$\left[f(x_1^D, x_2^D) - f\left(x_1^A, x_2^A\right)\right] = E_{11} + E_{22}$$
$$\left[f(x_1^D, x_2^D) - f\left(x_1^A, x_2^A\right)\right] = E_{21} + E_{12}$$

- $E_{11} = \left[f\left(x_1^D, x_2^D\right) - f\left(x_1^A, x_2^D\right)\right]$, contribution of x1 changing from D to A, fix x2 at x_2^D
- $E_{12} = \left[f\left(x_1^D, x_2^A\right) - f\left(x_1^A, x_2^A\right)\right]$, contribution of x1 changing from D to A, fix x2 at x_2^A
- $E_1 = \frac{1}{2}(E_{11} + E_{12})$, which is the average of the two.

  Similarly, $E_2 = \frac{1}{2}(E_{21} + E_{22})$
- What happens in linear model with no interactions?
  - $f(\boldsymbol{x}) = b_0 + b_1 x_1 + b_2 x_2$
  - $E_1 = b_1\left(x_1^D - x_1^A\right)$



Illustrative Example

Illustrative Example

Variable Name	Description	Monotone in probability of default
Response: y = default indicator	$y = 1$ if account defaulted and $y = 0$ if it did not default	
Predictors		
x1 = avg bal cards *std*	Average monthly debt standardized: amount owed by applicant) on all of their credit cards over last 12 months	**N**
x2 = credit age *std*	Age in months of first credit product standardized: first credit cards, auto-loans, or mortgage obtained by the applicant	**Y = Decreasing**
x3 = pct over 50 uti	Percentage of open credit products (accounts) with over 50% utilization	**N**
x4 = tot balance *std*	Total debt standardized: amount owed by applicant on all of their credit products (credit cards, auto-loans, mortgages, etc.)	**N**
x5 = uti open card	Percentage of open credit cards with over 50% utilization	**N**
x6 = num acc 30d past due 12 months	Number of non-mortgage credit-product accounts by the applicants that are 30 or more days delinquent within last 12 months (Delinquent means minimum monthly payment not made)	**Y = Increasing**
x7 = num acc 60d past due 6 months	Number of non-mortgage credit-product accounts by the applicants that are 30 or more days delinquent within last 6 months	**Y = Increasing**
x8 = tot amount currently past due *log*	Total debt standardized: amount owed by applicant on all of their credit products – credit cards, auto-loans, mortgages, etc.	**Y = Increasing**
x9 = num credit inq 12 month	Number of credit inquiries in last 12 months. An inquiry occurs when the applicant's credit history is requested by a lender from the credit bureau. This occurs when a consumer applies for credit.	**Y = Increasing**
x10 = num credit card inq 24-month	Number of credit card inquiries in last 24 months. An inquiry occurs when the applicant's credit history is requested by a lender from the credit bureau. This occurs when a consumer applies for credit.	**Y = Increasing**

Simulated Data:

- 50,000 accounts
- Default or not in 18 months
- 10 predictors
- Distributions of predictors mimic bureau data

Fitted Model:

- Feedforward NN
- Constrained to be monotone in indicated variables

Correlation

x1 = avg bal cards std
x2 = credit age std flip
x3 = pct over 50 uti
x4 = tot balance std
x5 = uti open card
x6 = num acc 30d past due 12 months
x7 = num acc 60d past due 6 months
x8 = tot amount currently past due log
x9 = num credit inq 12 month
x10 = num credit card inq 24-month

- Block correlation among similar features
- High-levels

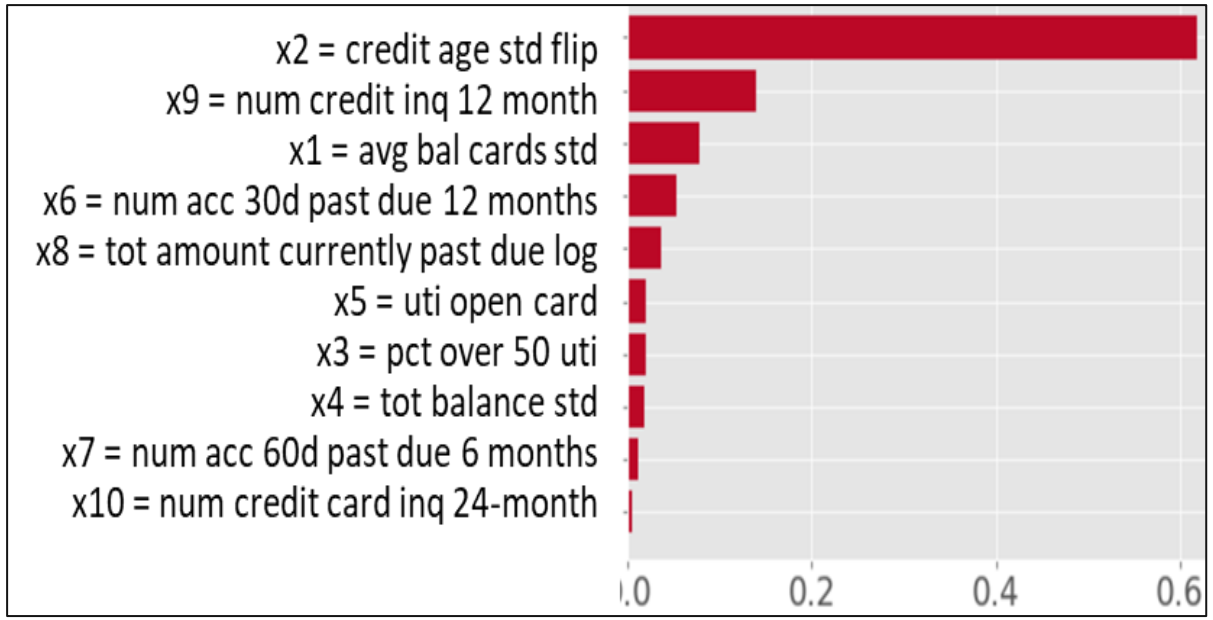
Training Monotone Neural Network

- Iterative algorithm: Fit with a penalty for monotonicity; certify; and iterate
- 50,000 accounts → training: 60%, validation and testing: 20% each
- Final model: three hidden layers with dimensions [35, 15, 5]; learning rate (LR) = 0.001
- For comparison:
  – fitted unconstrained Feedforward Neural Network (FFNN) [23, 35, 15]; LR = 0.004

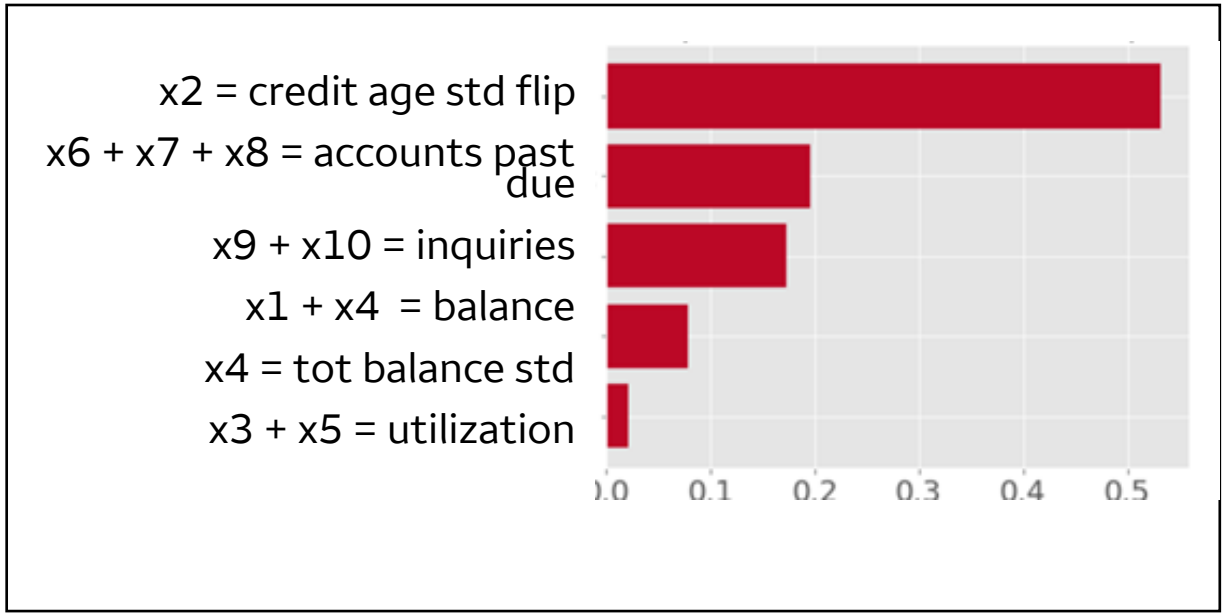
Training and Test AUCs for the Two Algorithms

Algorithm	Training AUC	Test AUC
FFNN	0.810	**0.787**
M-NN	0.807	**0.797**

Variable Importance for Mono-NN: All and Correlated

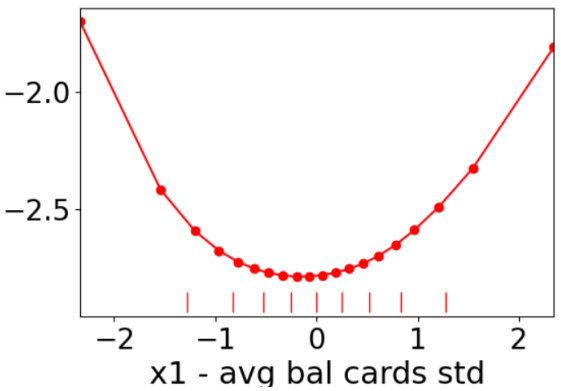


Individual Variable Importance

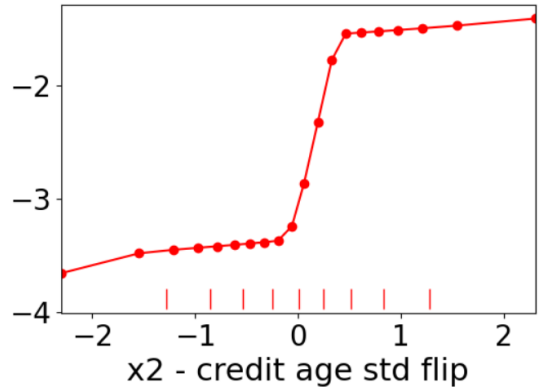


Joint Variable Importance for Correlated predictors

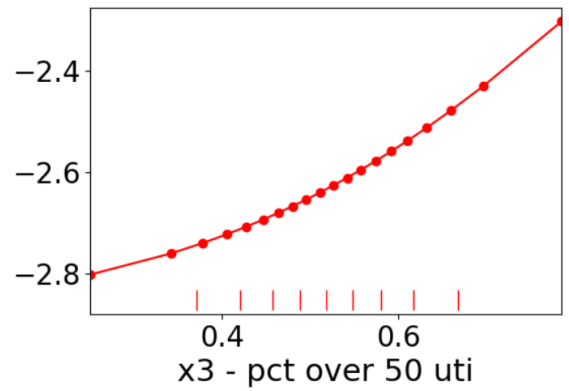
PDPs for Mono-NN



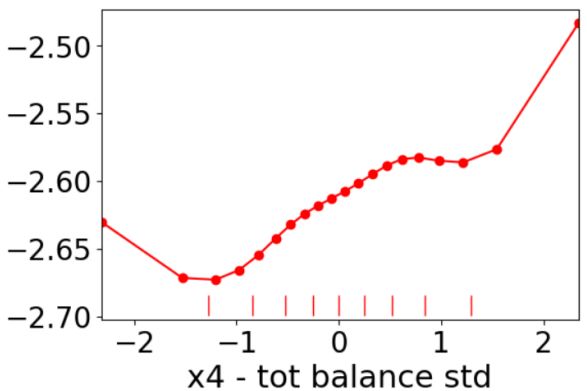
x1 - avg bal cards std



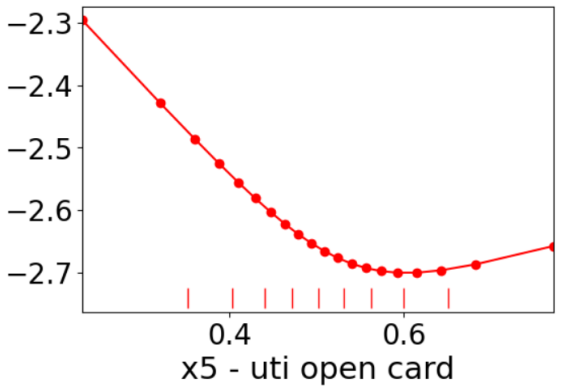
x2 - credit age std flip



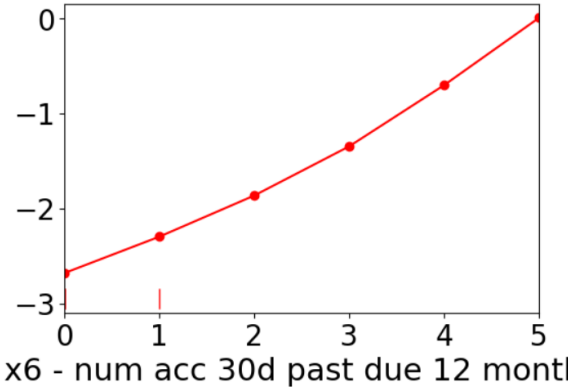
x3 - pct over 50 uti



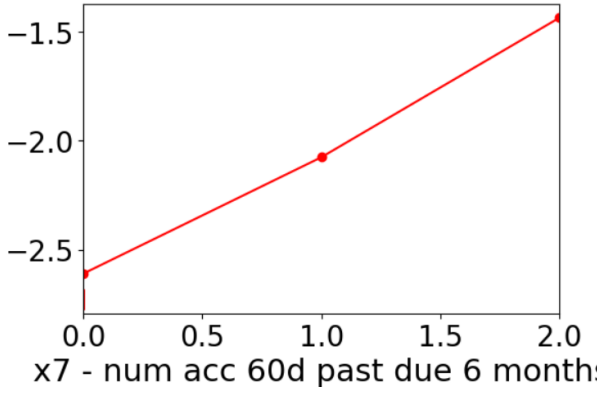
x4 - tot balance std



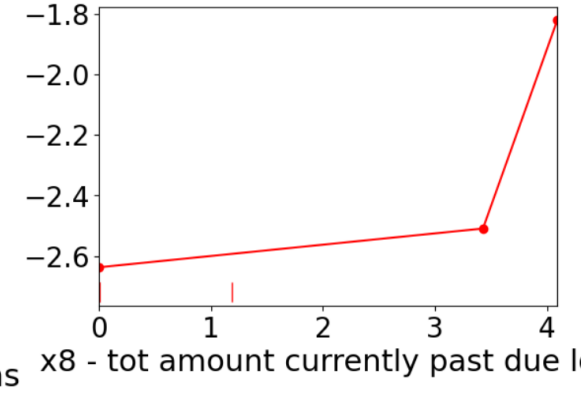
x5 - uti open card



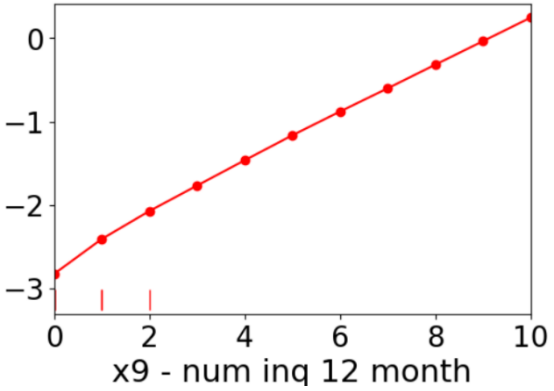
x6 - num acc 30d past due 12 month



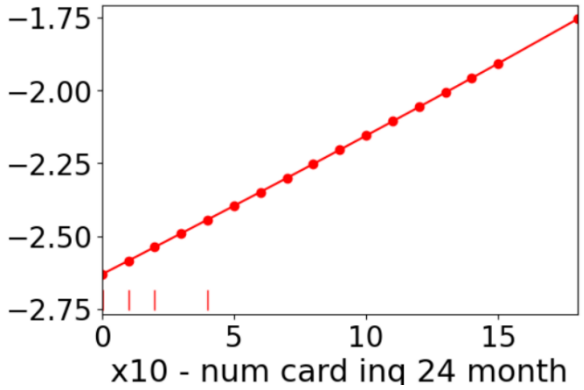
x7 - num acc 60d past due 6 months



x8 - tot amount currently past due log



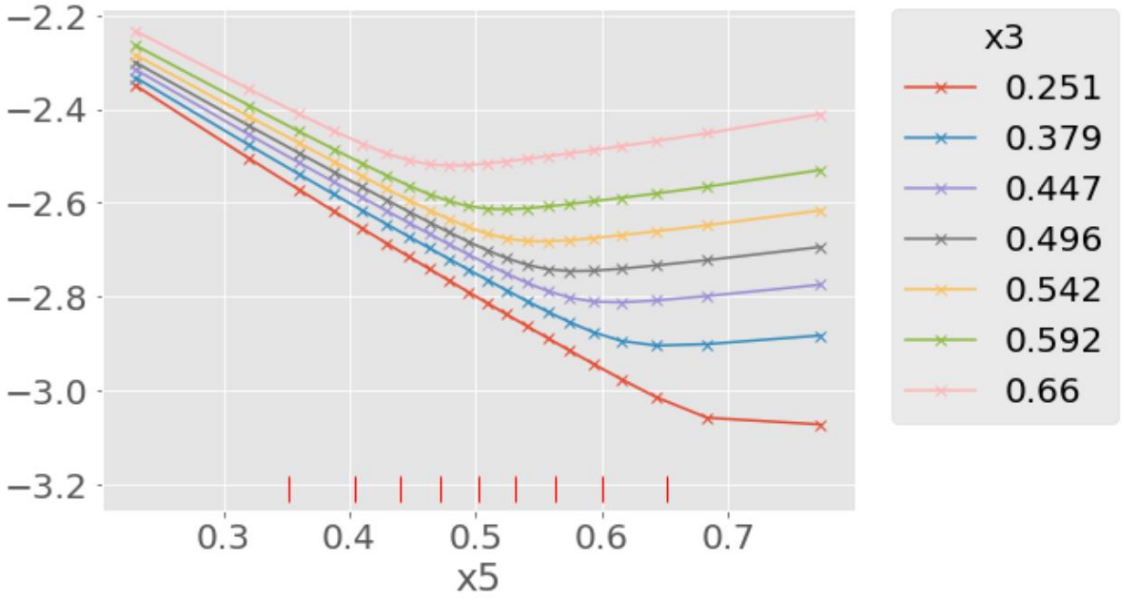
x9 - num inq 12 month



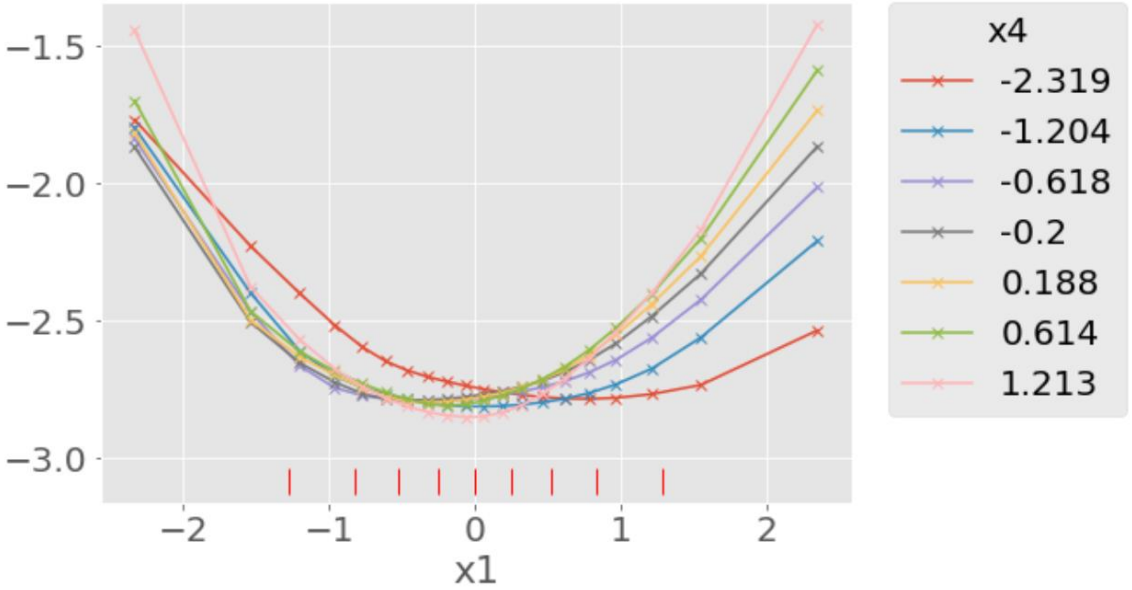
x10 - num card inq 24 month

Two-dimensional PDPs of Variables with Interactions

x5 = uti open card x3 = pct over 50 uti



x1 = avg bal cards std x4 = tot balance std



AA explanation: Decision rule – decline if $p(x) > 0.25$

Predictors	x^A	x_1^D	M-NN Attributions for x_1^D	x_2^D	M-NN Attributions for x_2^D
x1 = avg bal cards std	-0.006	0.674	0.112 (3.5%)	0.519	0.028 (0.5%)
x2 = credit age std flip	-0.733	0.886	**1.928 (59.5%)**	0.431	**1.565 (26.5%)**
x3 = pct over 50 uti	0.518	0.531	0.001 (0.0%)	0.522	-0.001 (0.0%)
x4 = tot balance std	-0.001	0.562	−0.008 (0.2%)	1.968	-0.201 (−3.4%)
x5 = uti open card	0.501	0.577	0.012 (0.4%)	0.525	-0.024 (−0.4%)
x6 = num acc 30d past due 12 months	0.000	0.000	0.0 (0.0%)	4.000	**1.850 (31.3%)**
x7 = num acc 60d past due 6 months	0.000	0.000	0.0 (0.0%)	2.000	**0.984 (16.6%)**
x8 = tot amount currently past due std	0.000	0.000	0.0 (0.0%)	4.379	**1.712 (28.9%)**
x9 = num credit inq 12 month	0.000	3.000	**1.010 (31.2%)**	0.000	0.0 (0.0%)
x10 = num credit inq 24 month	0.000	4.000	0.186 (5.7%)	0.000	0.0 (0.0%)
$\hat{p}(x)$	0.016	0.294		0.858	
$f(x) = logit(\hat{p}(x))$	−4.117	−0.876		1.797	

AA explanation: Decision rule – decline if $p(x) > 0.25$

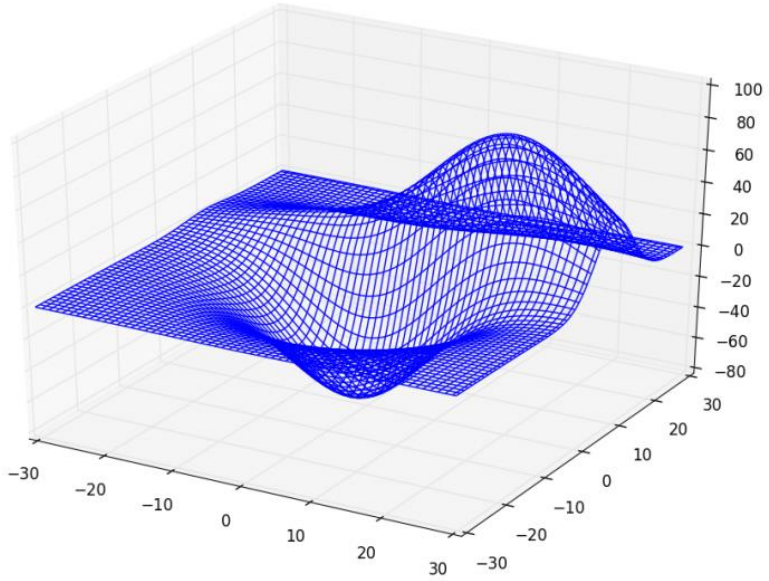
Predictors	x^A	x_1^D	M-NN Attributions for x_1^D
x1 = avg bal cards std	-0.006	0.674	0.112 (3.5%)
x2 = credit age std flip	-0.733	0.886	**1.928 (59.5%)**
x3 = pct over 50 uti	0.518	0.531	0.001 (0.0%)
x4 = tot balance std	-0.001	0.562	−0.008 (0.2%)
x5 = uti open card	0.501	0.577	0.012 (0.4%)
x6 = num acc 30d past due 12 months	0.000	0.000	0.0 (0.0%)
x7 = num acc 60d past due 6 months	0.000	0.000	0.0 (0.0%)
x8 = tot amount currently past due std	0.000	0.000	0.0 (0.0%)
x9 = num credit inq 12 month	0.000	3.000	**1.010 (31.2%)**
x10 = num credit inq 24 month	0.000	4.000	0.186 (5.7%)
$\hat{p}(x)$	0.016	0.294	
$f(x) = logit(\hat{p}(x))$	−4.117	−0.876	

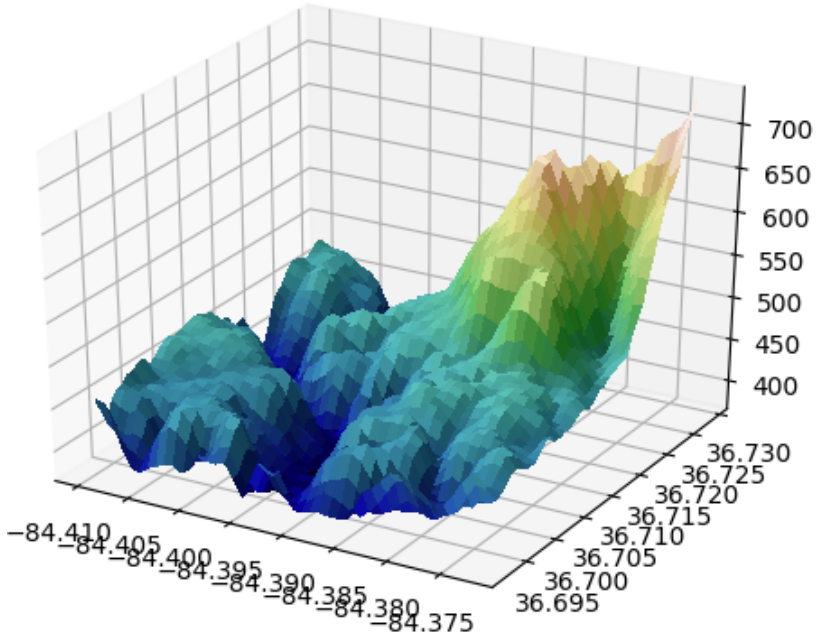
- Can modify to get combined explanation for groups of correlated predictors

Groups of predictors	M-NN Attributions for x_1^D
balance	0.126 (3.9%)
credit age std flip	**1.925 (59.4%)**
utilization	0.018 (0.5%)
num acc	0.000 (0.0%)
num inq	**1.173 (36.2%)**

Summary

- ML models have good predictive performance but hard to understand. Understanding the ML model is important to make sure the model is conceptually sound.
- **Most post-hoc tools** for studying input-output relationships are **lower-dimensional summaries**
  - **Limited** in **ability** to **characterize complex models** that may have different local behaviors
  - **Need better visualization** tools in high-dimensions





- Correlation can create havoc.
- Inherently Interpretable model can be both performant and interpretable (next class).